

DETAILED ACTION

In response to Applicant's Response After NonFinal Office Action filed 04/18/2008, and telephone interview on 04/17/2008, the examiner's amendment was authorized by attorney of record Christopher J. Lutz, Attorney for Applicants, Registration No. 44,883 at Chapin Intellectual Property Law, LLC Westborough Office Park 1700 West Park Drive Westborough, Massachusetts .01581 Telephone: (508) 616-9660.

Claims 34, 35 and 37 are currently amended. Claims 3, 10, 12, 17, 19-21, 24 and 26 were previously cancelled. Claims 1-2, 4-8, 11, 13-16, 18, 25, 27-33 and 36 were previously presented. Claims 9 and 22-23 were original.

Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

EXAMINER'S AMENDMENT

The application has been amended as follows:

1. (Previously Presented) A method for producing an output report comprising:
identifying a page template indicative of an output report having passive content,
the passive content being static;

parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a javabeen component such that parsing further comprises identifying the javabeen component, the javabeen component operable to process the metadata token;

retrieving, from a metadata repository, by the javabeen component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens; and

processing the metadata component by the javabeen component;

for each token, generating the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be retrieved from a dynamic repository and the display data operable to indicate organization of the fetched output data; generating the dynamic content further comprising:

retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report;

processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens; and

building the output report by assembling the generated dynamic content for each token in the page template, the page template indicative of a plurality of output reports,

the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report, building further comprising rendering an output page by: fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation;

retrieving the dynamic content from the fetched page descriptor; and

inserting the dynamic content in the output report by replacing the metadata tokens.

2. (Previously Presented) The method of claim 1 further comprising receiving a user request for an HTML page, wherein said building further comprises rendering the HTML page responsive to the user request.

3. (Canceled)

4. (Previously Presented) The method of claim 1 wherein said building the output report further comprises

receiving an HTTP request on behalf of a user, the HTTP request indicative of an HTML response;

determining the page template corresponding to the requested HTML response;

generating, in a rendering engine, a JSP output page from the determined page template; and

transmitting the generated JSP output page to the user as an HTML page.

5. (Previously Presented) The method of claim 1 wherein the display data is formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data.

6. (Previously Presented) The method of claim 5 wherein the predetermined syntax is a different syntax than the page template and said generating further comprises interpreting the display data from the predetermined syntax into the syntax defining the page template.

7. (Previously Presented) The method of claim 5 wherein the predetermined syntax includes syntactical elements, and is further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure.

8. (Previously Presented) The method of claim 7 wherein the dynamic content is active content and the predetermined syntax conforms to XML, in which the syntactical elements further comprise XML tags.

9. (Original) The method of claim 8 wherein the predetermined syntax is a page descriptor syntax defined by an XML schema.

Art Unit: 2176

10. (Canceled)

11. (Previously Presented) The method of claim 1 wherein the metadata components further comprise page descriptors, the page descriptors conforming to a declarative syntax and indicative of the dynamic content.

12. (Canceled)

13. (Previously Presented) The method of claim 11 wherein the metadata component is an XML file including at least one page descriptor according to a predetermined syntax, and said processing further comprises retrieving the at least one page descriptors by the javabeen component and producing HTML code corresponding to the at least one page descriptors.

14. (Previously Presented) The method of claim 1 wherein the display data further comprises user specific views, the user specific views indicative of formatting and display preferences specific to a particular user and operable for interpretation by syntax processing components to render the display data according to the user specific views.

15. (Previously Presented) A data processing device responsive to an encoded set of processor based instructions, the instructions encoded in a memory of the data processing device for producing an output report comprising:

a web server operable to identify a page template indicative of an output report having passive content, the passive content being static the page template indicative of a plurality of output reports;

a parser operable to parse tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, the metadata token corresponding to a javabean component and the parser further operable to identify the javabean component, the javabean component operable to process the metadata token, and the syntax processor is operable to invoke the javabean component for retrieving and processing the metadata component;

a syntax processor operable to, for each token, generate the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be fetched from a dynamic repository and the display data operable to indicate organization of the fetched output data, the syntax processor further operable to:

retrieve, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report; and

process the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens, the syntax processor further operable to process the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data,

the predetermined syntax being a different syntax than the page template and said generating further comprising interpreting the display data from the predetermined syntax into the syntax defining the page template, wherein the predetermined syntax includes syntactical elements, and is further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure; and

a formatter operable to build the output report by assembling the generated dynamic content for each token in the page template, the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report, building by the formatter further comprising rendering an output page by:

fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation;

retrieving the dynamic content from the fetched page descriptor; and

inserting the dynamic content in the output report by replacing the metadata tokens.

Art Unit: 2176

16. (Previously Presented) The data processing device of claim 15 wherein the web server further comprises an interface operable to receive a user request for an HTML page, wherein said building further comprises rendering the HTML page responsive to the user request.

17. (Canceled)

18. (Previously Presented) The data processing device of claim 15 wherein the formatter is further operable to produce the output report by:

receiving an HTTP request on behalf of a user, the HTTP request indicative of an HTML response;

determining the page template corresponding to the requested HTML response;

generating, in a rendering engine, a JSP output page from the determined page template; and

transmitting the generated JSP output page to the user as an HTML page.

Claims 19-21. (Canceled)

22. (Original) The data processing device of claim 15 wherein the dynamic content is active content and the predetermined syntax conforms to XML, in which the syntactical elements further comprise XML tags.

23. (Original) The data processing device of claim 15 wherein the predetermined syntax is a page descriptor syntax defined by an XML schema.

24. (Canceled)

25. (Previously Presented) The data processing device of claim 15 wherein the metadata components further comprise page descriptors, the page descriptors conforming to a declarative syntax and indicative of the dynamic content.

26. (Canceled)

27. (Previously Presented) The data processing device of claim 25 wherein the metadata component is an XML file including at least one page descriptor according to the predetermined syntax, and the syntax processor is further operable to retrieve the page descriptors via the javabeen components and produce HTML code corresponding to the at least one page descriptor.

28. (Previously Presented) The data processing device of claim 15 wherein the display data further comprises user specific views, the user specific views indicative of formatting and display preferences specific to a particular user and operable for interpretation by syntax processing components to render the display data according to the user specific views.

29. (Previously Presented) A computer program product having a computer readable medium storing computer program logic embodied in computer program code encoded thereon for producing an output report comprising:

computer program code for identifying a page template indicative of an output report having passive content, the passive content being static, the page template indicative of a plurality of output reports;

computer program code for parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a javabeen component such that parsing further comprises identifying the javabeen component, the javabeen component operable to process the metadata token;

retrieving, from a metadata repository, by the javabeen component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens; and

processing the metadata component by the javabeen component;

computer program code operable for generating, for each parsed token, the dynamic content, said generating further comprising fetching output data and processing display data, the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data, the output data operable to be retrieved from a dynamic repository and the display data

operable to indicate organization of the retrieved output data, the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report, generating the dynamic content further comprising:

- retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report;

- processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens;
- and

- computer program code for building the output report by assembling the generated dynamic content for each token in the page template, building further comprising rendering an output page by:

- fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation;

- retrieving the dynamic content from the fetched page descriptor, wherein the predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into a syntax of the output page; and

- inserting the dynamic content in the output report by replacing the metadata tokens;

computer program code for invoking the javabeen components for each of the metadata components to perform processing using the retrieved display data and the output data; and

computer program code for generating the dynamic content corresponding to each of the parsed metadata tokens.

30. (Previously Presented) An encoded set of processor based instructions having program code in an executable form on a computer readable storage medium for producing an output report comprising:

program code for identifying a page template indicative of an output report having passive content, the passive content being static, the page template indicative of a plurality of output reports;

program code for parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a javabeen component such that parsing further comprises:

identifying the javabeen component, the javabeen component operable to process the metadata token; and

retrieving, from a metadata repository, by the javabeen component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens;

program code for generating, for each token, the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be fetched from a dynamic repository and the display data operable to indicate organization of the fetched output data, generating the dynamic content further comprising:

retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report;

processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens, the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data, the predetermined syntax being a different syntax than the page template and said generating further comprising interpreting the display data from the predetermined syntax into the syntax defining the page template, the predetermined syntax including syntactical elements, and further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure; and

program code for building the output report by assembling the generated dynamic content for each token in the page template, building further comprising rendering an output page by:

fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation; and

retrieving the dynamic content from the fetched page descriptor, the retrieved dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report; and

inserting the dynamic content in the output report by replacing the metadata tokens.

31. (Previously Presented) A processor based information processing device responsive to encoded set of processor based instructions encoded in a memory of the information processing device for producing an output report comprising:

means for identifying a page template indicative of an output report having passive content, the passive content being static, the page template indicative of a plurality of output reports;

means for parsing tokens from the page template, the tokens being metadata tokens indicative of dynamic content, the dynamic content adapted to provide at least a portion of the output report, each of the metadata tokens corresponding to a javabeen component such that the means for parsing further comprises:

means for identifying the javabeen component, the javabeen component operable to process the metadata token; and

means for retrieving, from a metadata repository, by the javabeen component, metadata components corresponding to the metadata tokens and operable to provide the dynamic content corresponding to the parsed metadata tokens; and

processing the metadata component by the javabean component;

means for generating, for each token, the dynamic content, said generating further comprising fetching output data and processing display data, the output data operable to be fetched from a dynamic repository and the display data operable to indicate organization of the fetched output data, generating the dynamic content further comprising:

retrieving, based on the metadata components, the display data and the output data from a repository, the display data and output data corresponding to the dynamic content for rendering on the output report; and

processing the metadata components using the retrieved display data and output data to generate the dynamic content corresponding to the parsed metadata tokens, the display data formatted according to a predetermined syntax, the predetermined syntax operable for parsing and verifying the display data, the predetermined syntax being a different syntax than the page template and said generating further comprising interpreting the display data from the predetermined syntax into the syntax defining the page template, the predetermined syntax including syntactical elements, and further operable for nesting the syntactical elements, the nested syntactical elements defining a hierarchical structure; and

means for building the output report by assembling the generated dynamic content for each token in the page template, means for building further comprising means for rendering an output page by:

means for fetching, based on the parsed token, a page descriptor corresponding to the parsed token, the page descriptor defining the metalanguage representation;

means for retrieving the dynamic content from the fetched page descriptor and means for inserting the dynamic content in the output report by replacing the metadata tokens, the dynamic content further comprising java server pages referencing metalanguage representations of at least portions of other java server pages corresponding to the same rendered output report, the predetermined syntax comprising javabeen components operable to interpret the predetermined syntax into the syntax of the output page, wherein the means for generating further comprises: means for invoking the javabeen components for each of the metadata tokens to perform processing using the display data and the fetched output data; and means for generating the dynamic content corresponding to each of the parsed tokens.

32. (Previously Presented) The data processing device of claim 15 wherein the predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into a syntax of the output page, wherein the syntax processor: invokes the javabeen components for each of the tokens to perform processing using the display data and the output data; and generates the dynamic content corresponding to each of the parsed tokens.

33. (Previously Presented) The data processing device of claim 32 wherein the syntax processor is further operable to invoke the javabeen components to interpret the predetermined XML compliant syntax of a metadata component corresponding to the tokens for display on a user device.

34. (Currently Amended) The method of claim 1 wherein the predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into a syntax of the output page, further comprising:
invoking the javabeen components for each of the tokens to perform processing using the display data and the output data; and
generating the dynamic content corresponding to each of the parsed tokens.

Deleted: data processing device

35. (Currently Amended) The method of claim 34 wherein further comprising
invoking the javabeen components to interpret the predetermined syntax, the syntax being XML compliant, of a metadata component corresponding to the token for display on a user device.

Deleted: data processing device

36. (Previously Presented) The method of claim 1 wherein a predetermined syntax comprises javabeen components operable to interpret the predetermined syntax into the syntax of the output page, further comprising
invoking the javabeen components for a metadata component corresponding to the token to perform processing using the display data and the output data; and

generating the dynamic content corresponding to each of the parsed tokens.

37. (Currently Amended) The method of claim 36 further comprising invoking the javabeen components to interpret the predetermined syntax, the predetermined syntax being XML compliant, of the metadata components for display on a user device.

Deleted: data processing device

Response to Arguments

Applicant's arguments filed 04/18/2008 have been fully considered and they are persuasive.

Allowable Subject Matter

Claims 1-2, 4-9, 11, 13-16, 18, 22-23, 25 and 27-37 are allowed.

The following is an examiner's statement of reasons for allowance:

Interpreting the claims in light of the specification, Examiner finds the claimed invention is patentably distinct from the prior art of record, Hefetz et al. US 20040123238A1 filed 07/28/2003, in view of Ehring et al. US 20050097008A1 Continuation of 09/466,541 - filed 12/17/1999, in view of Hutsch et al US 20010034771A1 filed 01/12/2001, which set forth in the previous rejection mailed on 09/10/2007.

Under the broadest reasonable interpretation of the claimed limitation consistence with the Applicant's Specification, the prior art of record fail to teach all of

the Applicant's claimed limitation. In particular, the claimed invention advantageously provides a finer level of detail that invoking JavaBeans to produce JavaServer Pages (JSP) syntax from the parsed XML descriptors to pointed at by the parsed tokens (page 15, lines 6-27), and then assembles the dynamic portions to complete the page rendering that is producing the out put report page included the dynamic content at the web server, disclosed at page 16, lines 9-16 (see the remarks filed 04/18/2008 Pages 16-17.

In addition, It is noted, the original disclosure at page 18, Lines 17-21, stated, *"The operations and methods may be implemented in a software executable object or as a set of instructions embedded in a carrier wave,"* However, the examiner interpreted claims 29-30, as a computer program product having a computer readable medium storing computer program logic embodied in computer program code encoded thereon for producing an output report [hardware, machine], which is one of the conditions and requirements of 101 (process, machine, manufacture, or composition of matter). This interpretation is support by applicant's disclosure at Page 7, Lines 10-25, *" a computerized device such as a computer system, central processing unit, microprocessor, ... or other hardware device configured to process all of the method operations disclosed herein, a memory (e.g., any type of computer readable medium), to allow execution of instructions in a computer program such as a Java, HTML, XML, C, or C++ application"*. Thus claims 29-30 are statutory.

The Examiner asserts that the claims overcome the prior art of record when the limitations are read in combination with the respective claimed limitations in their

entirety.

The dependent claims, further limiting the independent claims, are also allowed.

Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance."

Conclusion

Any inquiry concerning this communication or earlier communications from the examiner should be directed to Quoc A. Tran whose telephone number is 571-272-8664. The examiner can normally be reached on Mon through Fri 8AM - 5PM.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Doug Hutton can be reached on (571)272-4137. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

Art Unit: 2176

/Quoc A. Tran/

Patent Examiner

/Doug Hutton/

Doug Hutton

Supervisory Primary Examiner

Technology Center 2100